2

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

OPTIMUM POSE MEASUREMENTS
FOR
KINEMATIC PARAMETERS IDENTIFICATION

by

Ronald L. Edwards

June, 1992

Thesis Advisor: Morris R. Driels

Approved for public release; distribution is unlimited

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified		1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY		3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			
4. PERFORMING ORGANIZATION REPORT NUMBER(S)		5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School	6b. OFFICE SYMBOL *(If applicable)* ME	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School	
6c. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000		7b. ADDRESS *(City, State, and ZIP Code)* Monterey, CA 93943-5000	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION	8b. OFFICE SYMBOL *(If applicable)*	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS *(City, State, and ZIP Code)*		10. SOURCE OF FUNDING NUMBERS	

Program Element No.	Project No.	Task No.	Work Unit Accession Number

11. TITLE *(Include Security Classification)*
OPTIMUM POSE MEASUREMENTS FOR KINEMATIC PARAMETERS IDENTIFICATION

12. PERSONAL AUTHOR(S) RONALD L. EDWARDS

13a. TYPE OF REPORT Master's Thesis	13b. TIME COVERED From To	14. DATE OF REPORT *(year, month, day)* JUNE 1992	15. PAGE COUNT 65

16. SUPPLEMENTARY NOTATION
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES

FIELD	GROUP	SUBGROUP

18. SUBJECT TERMS *(continue on reverse if necessary and identify by block number)*
ROBOTIC MANIPULATOR CALIBRATION

19. ABSTRACT *(continue on reverse if necessary and identify by block number)*

A six degree of freedom manipulator arm, a PUMA 560, is calibratied using random subsets of available experimental calibration data.. Some of these subsets produce good results motivating the search for an optimum procedure which will use a small number of poses. Statistical analysis of the joint excursions and end effector position variation in both "good" and "bad" subsets of poses were conducted. No significant statistical differences between them was discovered. The condition number of the Jacobian matrix is investigated as a potential measure of the accuracy which may be obtained from the subset under consideration. The condition number thus obtained contained too much variability to be a reliable predictor of accuracy. A computer simulation was conducted using a numerical optimizer to select the joint angles to be used for calibration. The optimizer studies failed to find an optimum set of poses for calibration. The conclusion of these studies is that there is no optimum set of poses to be used for calibration. An alternative hypthesis, that the resultant calibration accuracy depends only upon the accuracy of the measurements taken, seems to be proven.

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified	
22a. NAME OF RESPONSIBLE INDIVIDUAL M.R. Driels	22b. TELEPHONE *(Include Area code)* (408) 646-3383	22c. OFFICE SYMBOL ME/Dr

83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

Optimum Pose Measurements for Kinematic Parameters Identification

by

Ronald L. Edwards
Lieutenant Commander, United States Navy
B.S., Upper Iowa University, 1977

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1992

Author: Ronald L. Edwards
Ronald L. Edwards

Approved by: Morris Driels
Morris R. Driels, Thesis Advisor

Anthony J. Healey, Chairman
Department of Mechanical Engineering

ABSTRACT

A six degree of freedom robot manipulator arm, a PUMA 560, is calibrated using random subsets of available experimental calibration data. Some of these subsets produce good calibration results motivating the search for an optimum calibration procedure which will use a small number of poses. Statistical analysis of the joint excursions and end effector position variation in both "good" and "bad" subsets of poses were conducted. No significant statistical differences between them was discovered. The condition number of the Jacobian matrix is investigated as a potential measure of the accuracy which may be obtained from the subset under consideration. The condition number thus obtained contained too much variability to be a reliable predictor of accuracy. A computer simulation was conducted using a numerical optimizer to select the joint angles to be used for calibration. The optimizer studies failed to find an optimum set of poses for calibration. The conclusion of these studies is that there is no optimum set of poses to be used for calibration. An alternative hypothesis, that the resultant calibration accuracy depends only upon the accuracy of the measurements taken, *seems to be proven.*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

The objective of this thesis was to investigate an optimum method of calibration of robotic manipulators. Calibration of manipulators seeks to improve their accuracy. Accuracy is measured in terms of both position and orientation (called the pose) of the manipulator end effector. It is the difference between the commanded pose and the achieved pose of the end effector. The achieved pose of the manipulator is a function of fixed geometric properties, such as link lengths, and of variable geometric properties, such as the angular displacement in revolute joints. A kinematic model is developed for the manipulator using both the fixed and variable geometric data. Errors between the pose predicted by the model and the pose measured in the laboratory for a typical manipulator have been determined experimentally to be 10 mm or more [Ref. 1]. These errors arise due to the differences between the nominal values and as built values of the geometric properties of the manipulator. Improving the accuracy requires a method of accurately determining these parameters. Several practical methods of calibrating manipulators have been investigated [Ref. 2].

Repeatability is another performance measure of a manipulator. Repeatability is the average measure of how closely the manipulator can achieve a pose which has been previously taught. Experiments have shown that the repeatability of a typical manipulator is on the order of 0.3 mm [Ref. 3]. Successful calibration of manipulators should be viewed as calibration which results in an accuracy which is close to the repeatability of the manipulator.

Four basic steps in manipulator calibration have been identified and are briefly described as follows:

- A closed chain kinematic model of the manipulator and measurement system is developed. During this process, identifiable parameters are determined and the measured quantity or quantities are specified. A set of error functions are derived from the difference in the measured quantities and the quantities predicted by the model. Nominal parameter values are provided by the manipulator manufacturing specifications, measurement system specifications and the location of the measurement system.

- Next, experimental measurements are taken. These measurements are a function of the actual parameter values. Corresponding joint variable data is incorporated into the measurement set.

- Identification of the parameters is performed utilizing the experimental data. This process consists of systematically adjusting the nominal parameters until the model predictions match the experimental data and hence the error functions become zero.

- The final step involves incorporating the identified parameters into the software used to control the manipulator. [Ref. 4]

In previous work, Swayze [Ref. 5] performed the first three steps above on a PUMA 560 six degree of freedom manipulator arm and the fourth step was done in computer simulation. In his work, Swayze obtained calibration data consisting of 42 poses and the joint angles associated with them. This data was arbitrarily separated into two groups of 21 poses. The first group was used as an experimental group and the second group was used as a control or reference group. Using a computei program, TEST, random sets of six poses were selected from the experimental group and a manipulator calibration was performed. The kinematic parameters identified by this calibration were used with the joint angles of the control group poses to calculate predicted poses using a forward kinematic solution. These predicted poses were then compared with the actual poses which had been experimentally obtained and an average position error was calculated. Swayze found that some of these randomly selected sets of six poses produced small position errors which were close to the repeatability of the PUMA 560 manipulator. The best set of six poses produced a position error of 0.46 mm. Figure (1) is a flow chart of how the program TEST operates. Swayze's calibration data, which has not been previously published, is included as Appendix A. The program TEST was run for many iterations to investigate the range of position errors obtainable. The output files of program TEST were then screened by program SCREEN to select sets of poses which produced position errors less than 1.0 mm and greater than 20.0 mm. A summary of the output is tabulated in Appendix (B). These results raised several questions:

- What is unique about the sets of poses which produced small position errors?
- Can other small sets of poses be found which will also produce small position errors?
- Is there an optimum way to select a small number of poses which will yield a calibration with accuracy approaching the repeatability of a manipulator?

The answer to these questions is the topic of this thesis.

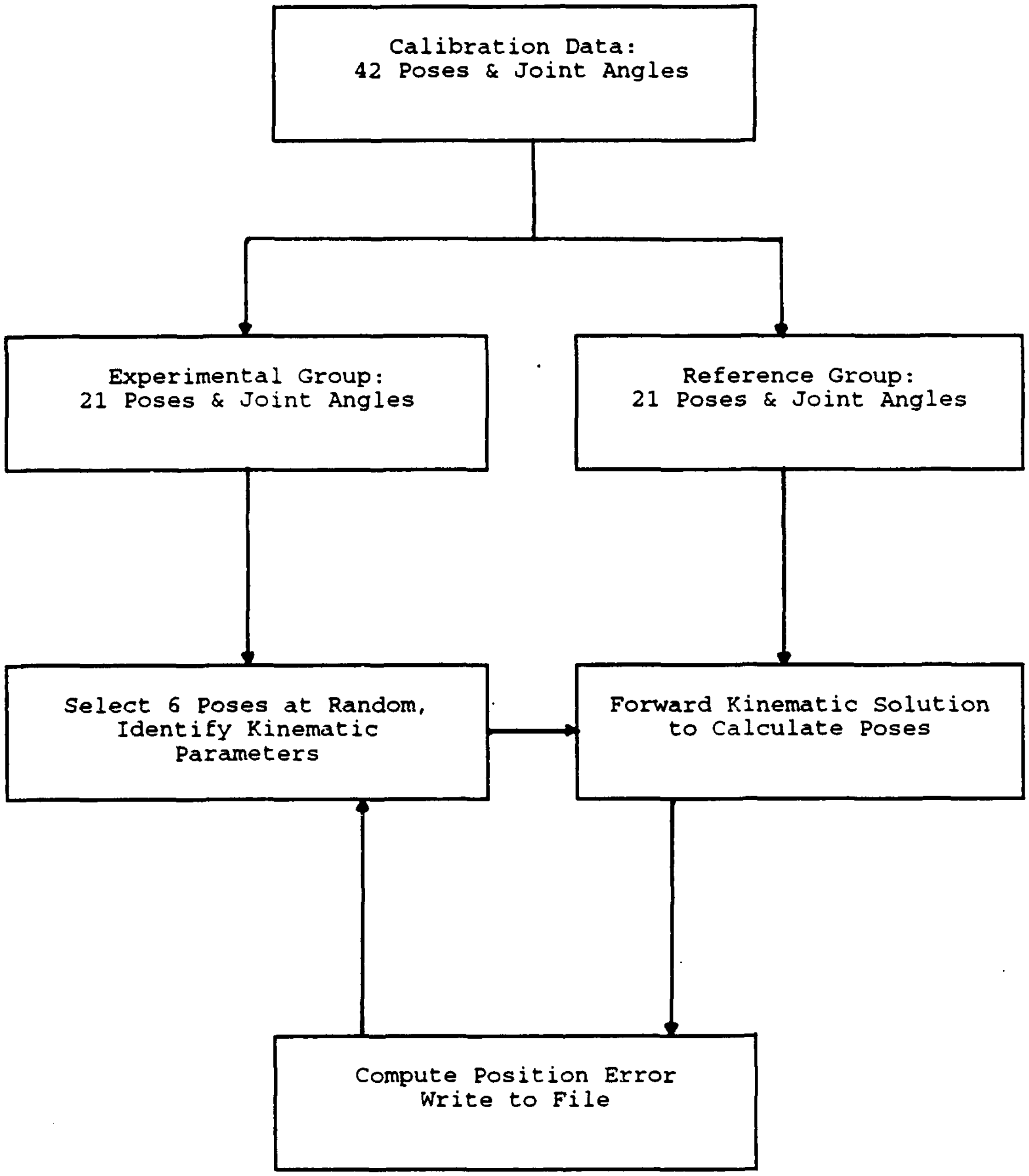


Figure 1. Flowchart for program TEST

II. THEORY

A. KINEMATIC MODELING OF MANIPULATORS

The theory presented and several diagrams of this chapter are based on material from Paul [Ref. 6]. The general arrangement of the material in this chapter closely follows the presentation of Swayze [Ref. 7].

1. General Coordinate System Transformations

Robotic manipulators are constructed of multiple links connected by either revolute or prismatic joints. The kinematic model of the manipulator consists of a Cartesian Coordinate frame attached to each link with a set of transformation equations to describe positions in one coordinate frame in terms of another coordinate frame.

Consider two coordinate frames, {0} and {1}, which have a common origin, where frame {1} is a produced from frame {0} by rotation of frame {0} by angle ψ about the x axis as shown in Figure (2). The position vector **P** may be represented in both frame {0} and frame {1} and in general will have different coordinates in each frame. Equations (1) are the transformation equations from frame {1} to frame {0} which can be readily seen by looking at a y-z planar projection of the two coordinate frames as shown in Figure (3).

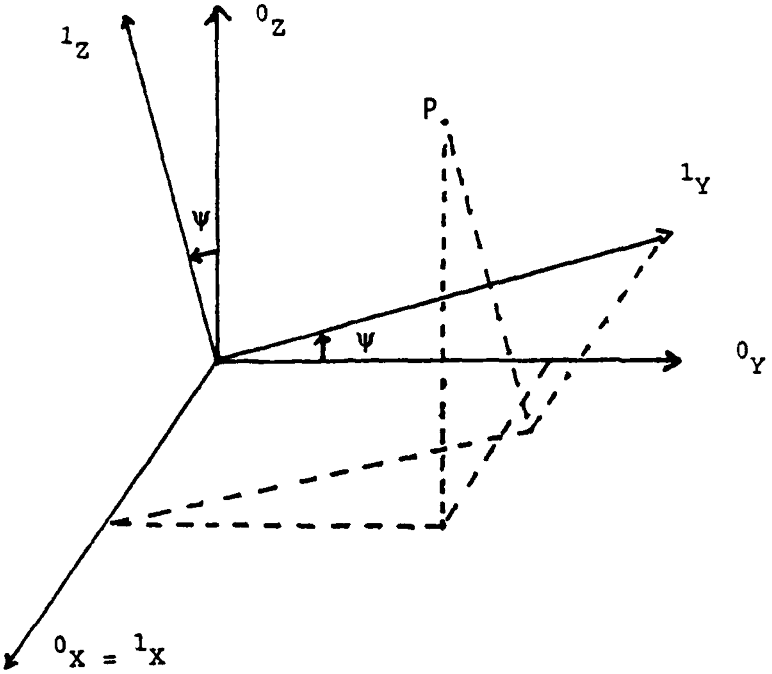


Figure 2. Rotation of coordinate frame about the x axis.

$$
\begin{aligned}
{}^0x_p &= {}^1x_p \\
{}^0y_p &= {}^1y_p\cos\psi - {}^1z_p\sin\psi \\
{}^0z_p &= {}^1y_p\sin\psi + {}^1z_p\cos\psi
\end{aligned}
\tag{1}
$$

Expressing equations (1) in matrix format results in

$$
\begin{bmatrix} {}^0x_p \\ {}^0y_p \\ {}^0x_p \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} \begin{bmatrix} {}^1x_p \\ {}^1y_p \\ {}^1z_p \end{bmatrix}
\tag{2}
$$

$$
or \qquad {}^0P = {}^0_1R\,{}^1P
\tag{3}
$$

where ${}^0_1R$ is the rotation matrix from frame {1} to frame {0}. Similar matrices can be derived for rotation about the y and z axes as well. For the remainder of this thesis, a

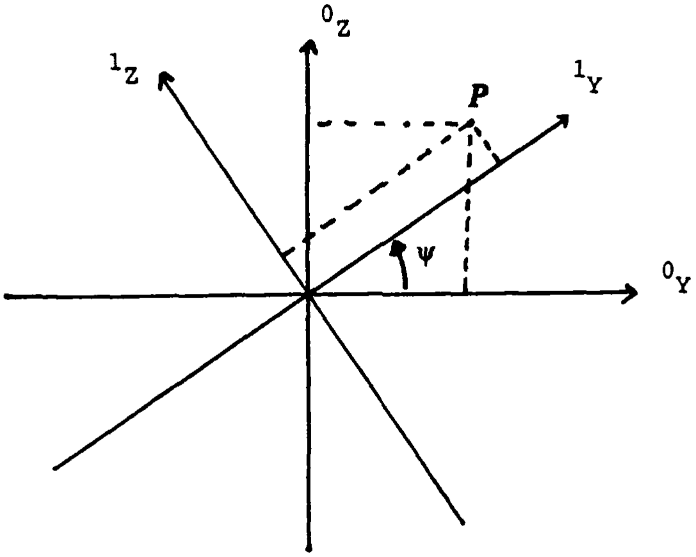


Figure 3. Coordinate frame rotation about the x axis projected onto the y-z plane.

rotation about a coordinate axis will be represented as Rot(x,ψ), Rot(y,θ), or Rot(z,φ). The matrices that represent these rotations are given by equations (4-6).

$$\boldsymbol{Rot(x,\psi)} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{bmatrix} \tag{4}$$

$$\boldsymbol{Rot(y,\theta)} = \begin{bmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{bmatrix} \tag{5}$$

$$\boldsymbol{Rot(z,\phi)} = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

Consider another coordinate frame, {2}, which is created by rotating frame {1} by an angle θ about the y axis of frame {1}. In this case the vector **P** can be represented in frame {1} as

$$ {}^{1}P = {}^{1}_{2}R\,{}^{2}P \qquad where \quad {}^{1}_{2}R = Rot(Y,\theta) \tag{7} $$

and in frame {0} as

$$ {}^{0}P = {}^{0}_{1}R\,{}^{2}P \qquad where \quad {}^{0}_{1}R = Rot(X,\psi) \tag{8} $$

or

$$ \begin{gathered} {}^{0}P = {}^{0}_{1}R\,{}^{1}_{2}R\,{}^{2}P = {}^{0}_{2}R\,{}^{2}P \\ where \quad {}^{2}_{0}R = {}^{0}_{1}R\,{}^{1}_{2}R. \end{gathered} \tag{9} $$

In the most general case, there can be translation as well as rotation of the coordinate frame. Let frame {1} be a translation from frame {0} by $[x\ y\ z]^T$ and frame {2} be a rotation of frame {1}. Then the position vector P will be expressed in frame {0} as:

$$ {}^{0}P = {}^{1}_{2}R\,{}^{2}P + \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{10} $$

It would be convenient to have both translations and rotations from one coordinate frame to another represented as a single matrix transformation. To allow this requires use of a 4 x 4 matrix where the upper left 3 x 3 sub-matrix is the same rotation matrix as before, the right hand column is the translation (x,y,z), and the last row of the matrix becomes [0 0 0 1]. This also requires augmented position vectors in the form

$$\boldsymbol{P} = \begin{bmatrix} P_x \\ P_y \\ P_z \\ 1 \end{bmatrix} \tag{11}$$

where the 1 is a scale factor which will always be 1 in this work. Thus, the complete transformation of rotation and translation can now be expressed as:

$${}^{1}_{0}T = \begin{bmatrix} n_x & o_x & a_x & P_x \\ n_y & o_y & a_z & P_y \\ n_z & o_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{12}$$

where the **n**, **o**, and **a** elements are the direction cosines of the x,y, and z axis of frame {1} (rotated frame) with respect to the original frame {0}. This 4 x 4 matrix is called the homogeneous transformation matrix and in general, represents a generalized rotation and translation of coordinate axes. If ${}^{0}_{1}T$ is the transformation matrix which transforms position vectors in frame {1} to frame {0} then

$${}^{0}P = {}^{0}_{1}T\,{}^{1}P. \tag{13}$$

If the position in the base frame {0} is known and the position in frame {1} is desired, then

$$\begin{aligned} {}^{1}P &= \left({}^{0}_{1}T\right)^{-1}\,{}^{0}P \\ &= {}^{1}_{0}T\,{}^{0}P\ . \end{aligned} \tag{14}$$

This presupposes that the inverse transformation matrix can be found. Rather than invert the **T** matrix, Paul [Ref. 8] describes the inverse as:

$$T^{-1} = \begin{bmatrix} n_x & n_y & n_z & -\mathbf{p}\cdot\mathbf{n} \\ o_x & o_y & o_z & -\mathbf{p}\cdot\mathbf{o} \\ a_x & a_y & a_z & -\mathbf{p}\cdot\mathbf{a} \end{bmatrix} \tag{15}$$

where **n**, **o**, **a**, and **p** are the column vectors of **T** and " · " represents the usual dot product of two vectors.

2. Roll, Pitch, Yaw and Euler Angle Transformations

A given orientation may be specified in terms of three rotations. There are 24 angle set conventions which may be used to specify these angles [Ref. 9]. In this work, rotations about fixed coordinate axes x-y-z will be used exclusively. This convention is referred to as roll, pitch and yaw (RPY) angles. The notation utilized will be

$$RPY(\phi,\theta,\psi) = Rot(z,\phi)Rot(y,\theta)Rot(x,\psi) \tag{16}$$

which is interpreted as a rotation about x by an angle ψ, followed by a rotation about y by an angle θ and lastly a rotation about z by an angle ϕ. Multiplying the rotation matrices for these 3 rotation together yields:

$$RPY(\phi,\theta,\psi) = \begin{bmatrix} c\phi c\theta & c\phi s\theta s\psi - s\phi c\psi & c\phi s\theta c\psi + s\phi s\psi & 0 \\ s\phi c\theta & s\phi s\theta s\psi + c\phi c\psi & s\phi s\theta c\psi - c\phi s\psi & 0 \\ -s\theta & c\theta s\psi & c\theta c\psi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

where s and c have been used for sine and cosine for brevity. Multiplying equation (17) by the translation matrix, Trans(x,y,z), shown in equation (18), will yield a complete homogeneous transformation. Pre-multiplication would be used when the x, y and z coordinates refer to the original coordinate axes prior to rotation and post-multiplication is used when the x, y and z coordinates are referenced to the rotated coordinate frame.

$$\mathbf{\mathit{Trans}}(x,y,z) = \begin{bmatrix} 1 & 0 & 0 & x \\ 0 & 1 & 0 & y \\ 0 & 0 & 1 & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (18)$$

Thus all transformations may thought of as a product of a rotation matrix and translation matrix of the form of equation (19).

$$\mathbf{\mathit{RPYT}} = \mathbf{\mathit{RPY}}(\phi,\theta,\psi)\mathbf{\mathit{Trans}}(x,y,z) \qquad (19)$$

Figure (4) shows a generalized transformation with roll, pitch and yaw rotations with a translation of x, y and z in the new coordinate directions.

3. Denavit-Hartenberg Transformations

As previously stated, three rotations and three translations are required, in general, to transform one coordinate system to another. In robotic manipulators the attachment of successive links imposes constraints and therefore fewer parameters are required to fully describe the transformation. Although there are many different types of manipulators, they are all constructed of links attached at joints. Several systematic methods of attaching coordinate frames to links have been established. One such method is from Denavit and Hartenberg (DH) [Ref. 10] and is described below.

Manipulator links are characterized by two parameters: link length a_n and twist angle α_n as shown in Figure (5). The link length a_n is defined as the length of the common normal between the axes for joints n and n+1. The twist angle α_n of a link is

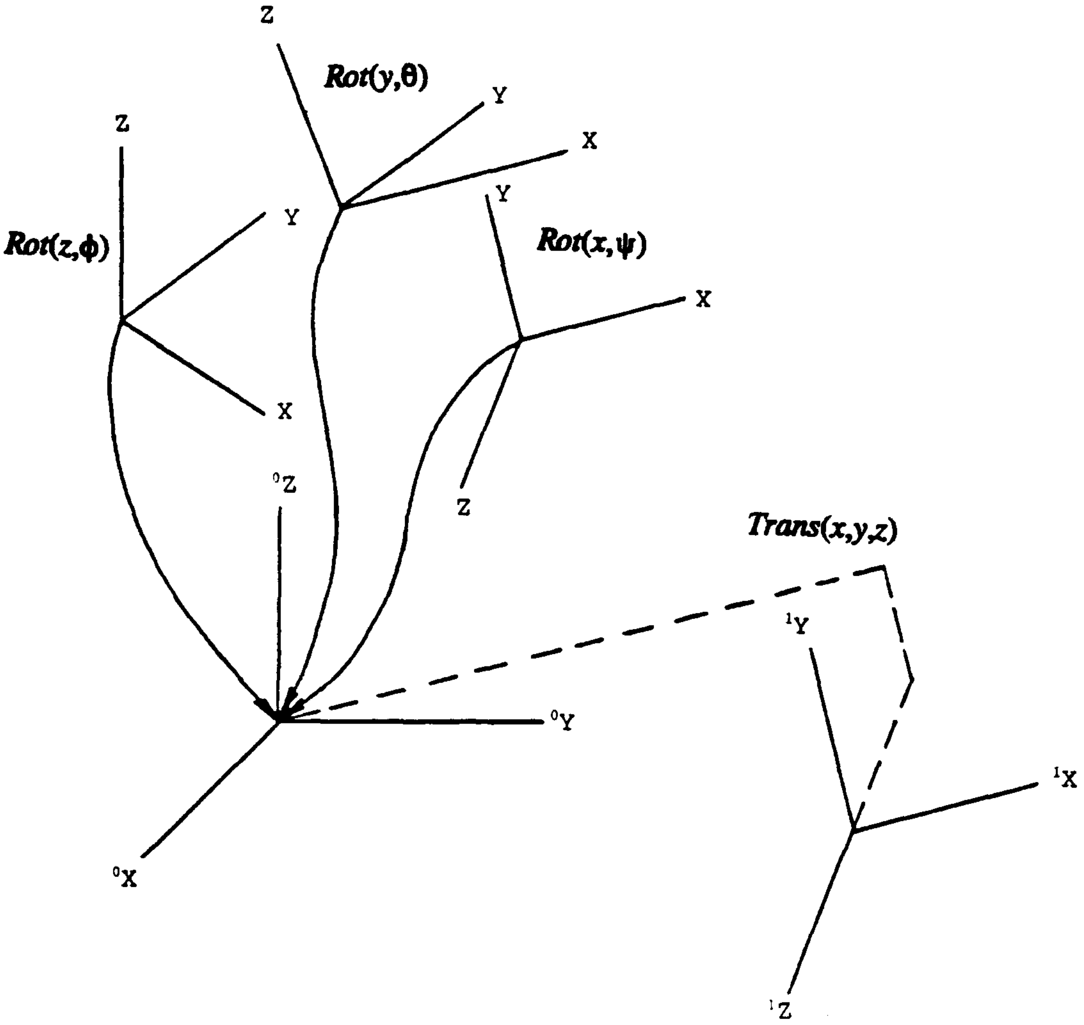


Figure 4. Generalized Roll, Pitch, Yaw, Translation Transformation

defined as the angle which joint axis n must be rotated about the common normal to reach joint axis n+1. A positive rotation is defined using the right hand rule considering the direction of travel from axis n to n+1 to be the positive direction. Thus it can be seen that the link length must always be greater than or equal to zero while the twist angle has no such restriction. The two links attached at a joint both have common normals as

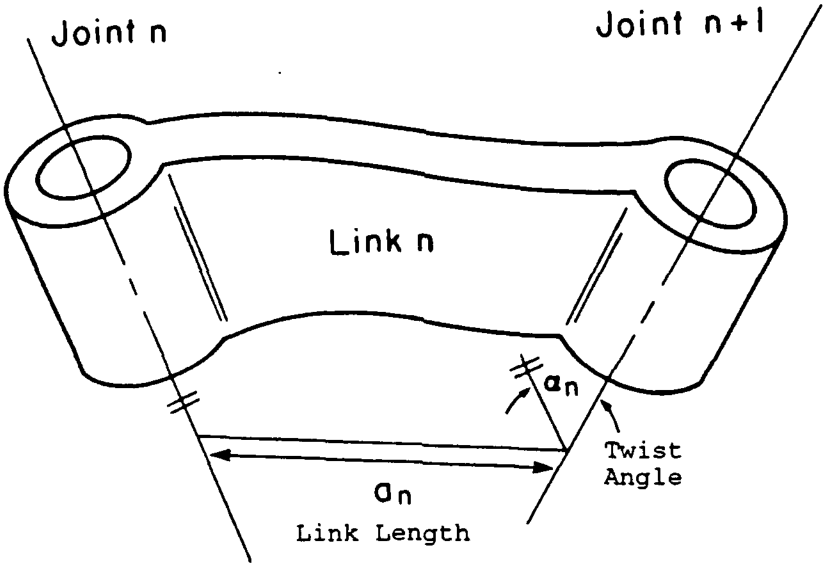


Figure 5. Link Length and Twist Angle

described above. The distance between these common normals, measured along joint axis n, is defined to be the joint offset d_n. The last parameter necessary to fully define the DH transformation is the joint angle θ_n. The joint angle is defined to be the angle between the two common normals of a joint measured in a plane normal to the joint axis.

Coordinate frames are established for each link of the manipulator by the following method. Link 0, which is the base of the manipulator, is attached to link 1, the first movable link, via joint 1. Link 2 is attached to link 1 via joint 2 and so on. The origin of coordinate frame {n}, associated with link n, is located at the point of intersection of the common normal joining joint axes n and n+1 and the axis of joint n+1. In the case of intersecting joint axes the origin is chosen to be the point of intersection. The Z axis of frame {n} is chosen to be the axis of joint n+1. The x axis is chosen to be collinear with the common normal with the positive direction for x being in the direction of travel from joint axis n to joint axis n+1. In the case of intersecting joint axes the x axis is chosen perpendicular to the plane of intersection of the two axes. With the x and z axes thus established the Y axis is dictated by the use of right hand coordinate systems. Note that the choice of the positive direction for z is arbitrary as is the positive direction of the x axis when the joint axes intersect. If joint axes are parallel then the position of the origin is chosen to make the link offset zero for the next link whose origin is defined. Figure (6) shows the assignment of link coordinate frames for a multiple link manipulator.

With coordinate frames established for each link the relationship between link n-1 and link n is established by two rotations and two translations as follows:

- rotate about z_{n-1} by an angle θ_n which establishes the correct direction for x_n;
- translate along z_{n-1} a distance d_n which gets the x_n axis in the correct location;
- translate along x_n a distance a_n which establishes the origin of frame {n} at the correct position;
- rotate about x_n by angle α_n which establishes the correct orientation for z_n and y_n.

These four steps are equivalent to equation (20) which is the DH transformation from frame {n-1} to frame {n}. Equation (21) is the matrix form of the DH transformation.

$$^{n}T = Rot(z,\theta)\, Trans(z,d)\, Trans(x,a)\, Rot(x,\alpha) \tag{20}$$

To describe a position vector from the end effector frame in the base frame is as easy as multiplying the position **P** by the **T** matrix for each link of the manipulator as shown in equation (22).

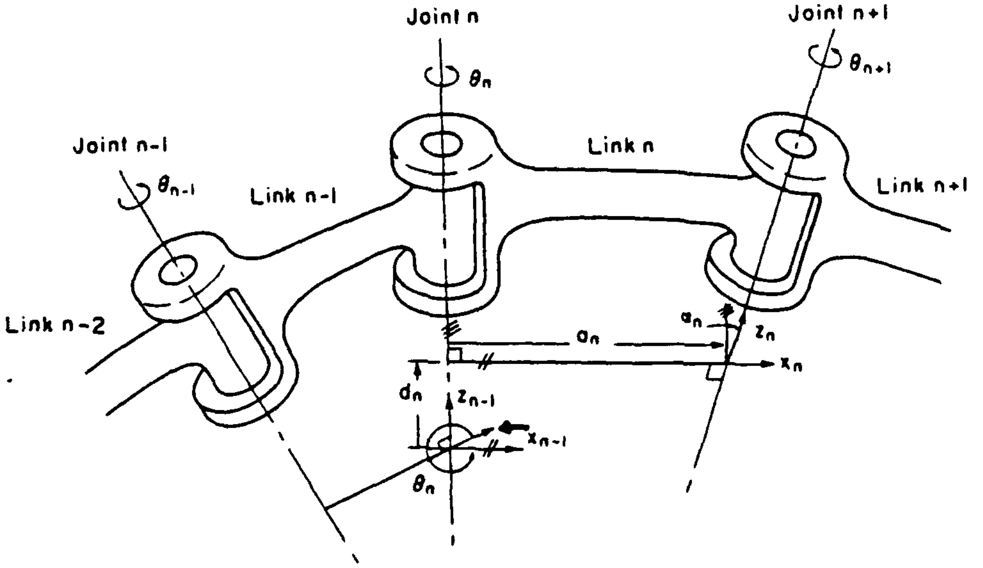


Figure 6. Assignment of link parameters θ, d, a, and α

$$
{}^{n}_{n-1}T = \begin{bmatrix} c\theta & -s\theta c\alpha & s\theta s\alpha & ac\theta \\ s\theta & c\theta c\alpha & -c\theta s\alpha & as\theta \\ 0 & s\alpha & c\alpha & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{21}
$$

$$
{}^{0}P = {}^{0}_{1}T\,{}^{1}_{2}T \ldots {}^{n-1}_{n}T\,{}^{n}P \tag{22}
$$

4. Modified Denavit-Hartenberg Transformations

For ideal calibrations of manipulators it is important that the kinematic model used be proportionate. That is, a small change in one kinematic property should result in only small changes in the other kinematic properties. In most cases the DH method results in a proportionate model. However, in the case of parallel or nearly parallel axes the method becomes disproportionate. This is readily apparent from Figure (7) which

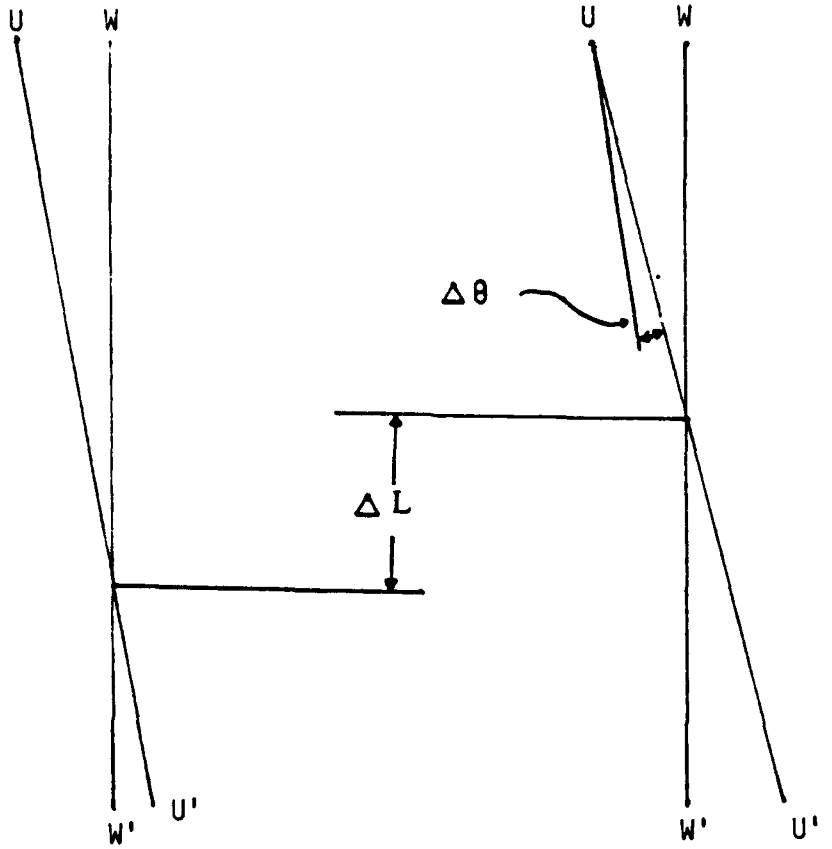


Figure 7. Disproportionate affects of small axis variations on manipulator calibration

represents two joints with nearly parallel intersecting axes. Let u-u′ represent the joint n-1 axis and w-w' represent the joint n axis. A small rotation of axis w-w′ will cause a large change in the point of intersection. In the case of non intersecting nearly parallel joint axes a small rotation in one of them will cause a large change in the length of the common normal a_n between them. This small rotation may even cause non-parallel axes to become parallel such that no unique common normal exists.

The modification to the standard DH transformations proposed by Hayati and Mirmirani [Ref. 11] will result in a proportionate model for consecutive revolute joints. The modification involves adding another rotation to the standard DH transformation and setting $d_n = 0$. The following transformation can be used to arrive at either the standard DH or the modified DH (MDH) transformation.

$${}^{n}_{n-1}T = Rot(z,\theta)\,Trans(z,d)\,Trans(x,a)\,Rot(x,\alpha)\,Rot(y,\beta) \tag{23}$$

Equation (24), which expresses equation (23) in matrix form, becomes the standard DH

$${}^{n}_{n-1}T = \begin{bmatrix} c\theta c\beta - s\theta s\alpha s\beta & -s\theta c\alpha & c\theta s\beta + s\theta s\alpha c\beta & ac\theta \\ s\theta c\beta + c\theta s\alpha s\beta & c\theta c\alpha & s\theta s\beta - c\theta s\alpha c\beta & as\theta \\ -c\alpha s\beta & s\alpha & c\alpha c\beta & d \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{24}$$

transformation by setting $\beta=0$ and becomes the MDH transformation by setting $d=0$. All future reference to transformations between links of manipulators will be of the form of equation (24). This facilitates a standard computer code which is used for both types of transformations.

5. Kinematic Chains and World Coordinate Frames

When a series of links is joined together to form a robotic manipulator the series of transformations from the end effector coordinate frame to the base coordinate frame can be thought of as a kinematic chain. In this chain each transformation is equivalent to one link of the manipulator. In general the base frame {0} is internal to the manipulator and therefore knowledge of the position of the tool with respect to the base frame is not very useful. A more useful reference coordinate frame would be the manipulator work space frame or world coordinate frame. This frame can be used to conduct measurements to be used for calibration and/or programming of the manipulator. Use of the world coordinate frame, designated frame {w}, requires use of an additional transformation to transform the base frame of the manipulator to world frame coordinates. This transformation may be done in one of two ways:

- Use RPYT(ϕ,θ,ψ,x,y,z) to transform frame {0} to frame {w}. This adds six parameters to the kinematic model.
- Use a DH style transformation to transform frame {0} to frame {w}. This requires only four parameters.

At first glance, the second method appears to require fewer parameters but this is not the case. The fallacy is in the fact that the frame {0} for each method is different. In the first case frame {0} will be coincident with frame {1} with the result that two of the parameters of the MDH transformation from frame{0} to frame {1} will be identically zero. In the second case four parameters will be required for the MDH transformation from frame {0} to frame {1}. Therefore, in both cases, a total of eight parameters will be required to transform between the world frame and the frame {1}. The last transformation in the chain is the transformation between the tool frame and frame {n-1} for an n link manipulator. This transformation will be the RPYT transformation. The total kinematic chain is shown in equation (25).

$$ {}^{w}_{e}T = {}^{w}_{1}T\, {}^{1}_{2}T \cdots {}^{n-2}_{n-1}T\, {}^{n-1}_{e}T \tag{25} $$

6. Application to the PUMA 560 Manipulator

The PUMA 560 manipulator consists of six links connected with revolute joints. The kinematic model is constructed using MDH transformations between the links and a RPYT transformation between link 5 and the end effector on link 6. The total number of parameters required for completeness is given by:

$$N = 4R + 2P + 6 \tag{26}$$

where N is the required number of independent parameters, R is the number of revolute joints, and P is the number of prismatic joints. [Ref. 12] The PUMA 560, with six revolute joints, thus requires 30 parameters for a complete model. A complete model is one which allows:

- the reference (world) coordinate frame to be arbitrarily selected;
- the zero position of each joint angle to be arbitrarily selected and;
- the tool coordinate frame to be arbitrarily attached to link six of the manipulator.

The MDH transformation has a specific definition of the zero position for each joint angle. To allow each joint zero position to be arbitrary a term $\delta\theta_i$ is introduced for each joint. Each θ_i is the angle as measured by the joint encoder and $\delta\theta_i$ is the encoder offset from the zero position established by the MDH procedure. Using this definition, θ_i for each joint is measured and considered fixed during a calibration while $\delta\theta_i$ is the parameter which is to be determined. $\delta\theta_i$ is considered to be constant for each joint throughout the range of travel of the joint angle. Table (1) shows the nominal kinematic parameters for the 30 parameter model. The values in parenthesis are defined to be zero (not part of the model) and the underlined values represent the placement of the reference coordinate frame and tool frame used during this work. Figure (8) shows the assignment of coordinate frames for the PUMA 560 manipulator.

B. PARAMETERS IDENTIFICATION METHODOLOGY

1. The Numerical Optimizer ZXSSQ

The IMSL subroutine ZXSSQ uses a Levenberg-Marquadt algorithm for the solution of non-linear least squares problems. The statement of the problem solved is given by equation (27). Each $f^i(x)$ is a user supplied objective function. The optimizer functions by calculating the gradient of the objective function at the current value of the design variable x using a finite difference approximation. It then changes x in a way that will reduce the objective function. This process is repeated until the objective function

Table 1. PUMA 560 NOMINAL KINEMATIC PARAMETER TABLE

T	$\delta\theta_i{}^\circ$	d_i (mm)	a_i (mm)	$\alpha_i{}^\circ$	$\beta_i{}^\circ$	
w→0	$\underline{180}$	$\underline{-394.6}$	$\underline{-404.5}$	$\underline{90}$	(0.0)	
0→1	$\underline{0.0}$	$\underline{473.6}$	0.0	-90	(0.0)	
1→2	0.0	(0.0)	432.1	0	0.0	
2→3	0.0	149.1	-19.2	90	(0.0)	
3→4	0.0	432.9	0.0	-90	(0.0)	
4→5	0.0	0.0	0.0	90	(0.0)	
RPYT	$\delta\phi$	$\delta\theta$	$\delta\psi$	P_x (mm)	P_y (mm)	P_z (mm)
5→e	$\underline{90}$	$\underline{0.0}$	$\underline{0.0}$	$\underline{0.0}$	$\underline{0.0}$	$\underline{132.2}$

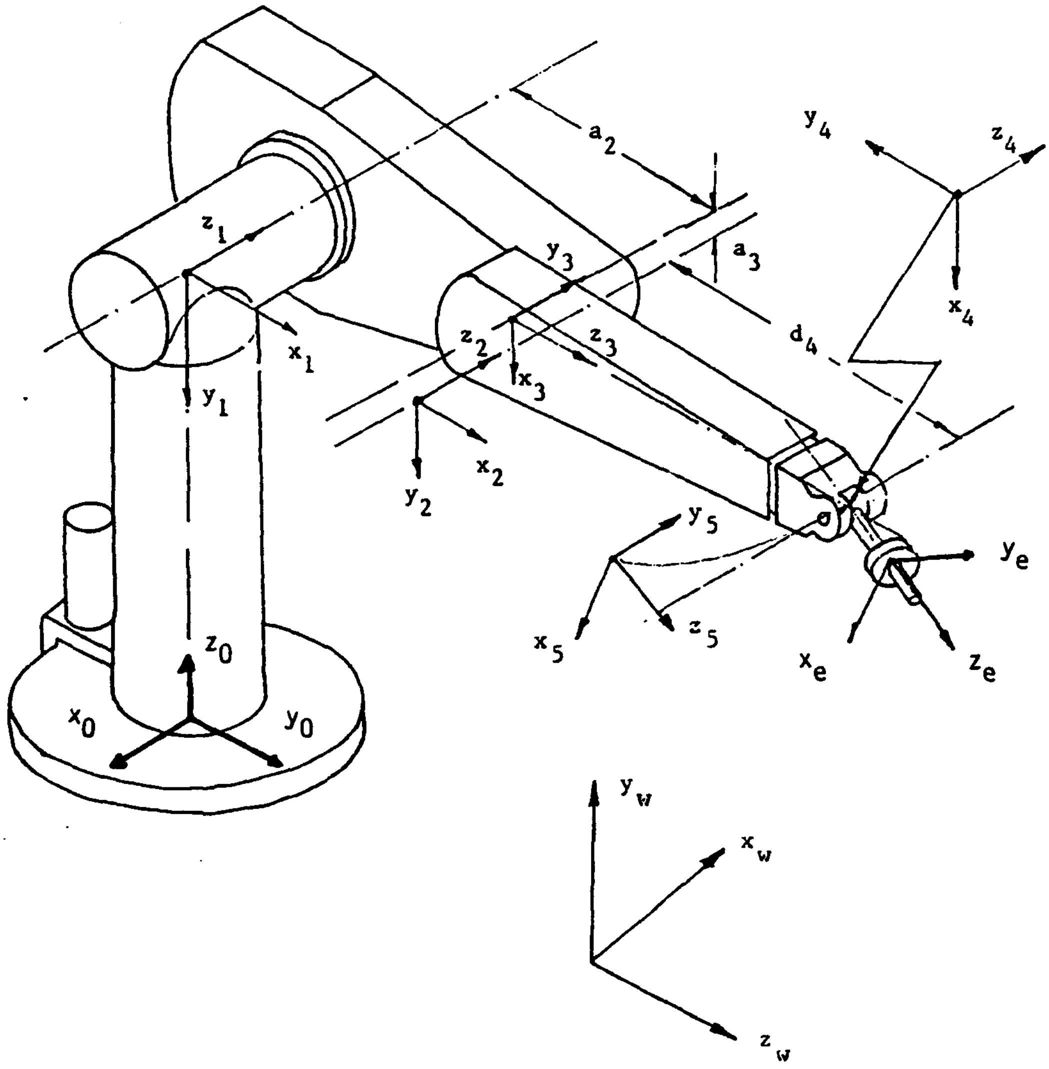


Figure 8. PUMA 560 Coordinate Frame Assignments using the DH Method

minimum has been found or the user specified maximum number of iterations has been reached. ZXSSQ then passes the calculated value of the solution vector x back to the calling program.

2. The Identification Jacobian Matrix

For each calibration experiment conducted the kinematic parameters identification algorithm calculates an identification Jacobian matrix which relates the calibration poses and the kinematic parameters by the equation (28),

$$\delta P = J\delta K \tag{27}$$

where δP is the difference between the experimentally measured pose and the calculated pose and δK is the difference between the calculated and the nominal kinematic parameters. J is the identification Jacobian matrix and will be calculated by the identification algorithm. In order to find δK we must find the inverse of J. In general J will not be a square matrix so the equation must be inverted by post-multiplying both sides by the pseudo-inverse yielding the following equation for δK.

$$\delta K = (J^T J)^{-1} J^T \delta P. \tag{28}$$

3. General Calibration Scheme

Given the previously introduced 30 parameter model for the PUMA 560, which is based on the nominal kinematic parameter values, the actual values of the parameters may be found by performing a calibration. The pose of the end effector is measured and the associated manipulator joint angles are recorded. The joint angles are changed and the new pose is measured and joint angles recorded as before. This process is repeated until a sufficient quantity of data is obtained. Each pose consists of both orientation (3 rotation angles) and position (3 translations) so a sufficient number of poses is 30/6=5. There is, of course, some noise in the measurement system so a number of poses greater than the sufficient number is used so that the effect of the measurement noise may be reduced. For each pose and associated joint angles a forward kinematic solution is calculated based on the nominal model. This is the calculated pose referred to in the previous section.

4. Calibration Implementation (Program ID6)

The FORTRAN program ID6 is used to calculate the PUMA 560 manipulator kinematic parameters. The program reads in the nominal values of the parameters from a file (INPUT.DAT) and the calibration data from another file (POSE.DAT). The program then calls the IMSL subroutine ZXSSQ which calls a subroutine PUMA_ARM. PUMA_ARM calculates the forward kinematic solution for each set of joint angles using the current best guess of the kinematic parameters which have been supplied by ZXSSQ in the calling statement. It then computes the difference between the calculated and measured poses. These differences are returned to ZXSSQ as the objective function values. ZXSSQ then changes the kinematic parameters and calls PUMA_ARM again. When the objective function is reduced to approximately zero (that is, the measured and calculated poses are nearly identical) ZXSSQ has identified the kinematic parameters which it then passes back to the calling program. ID6 then writes these values to an output file and terminates. Figure (9) is a block diagram of ID6.

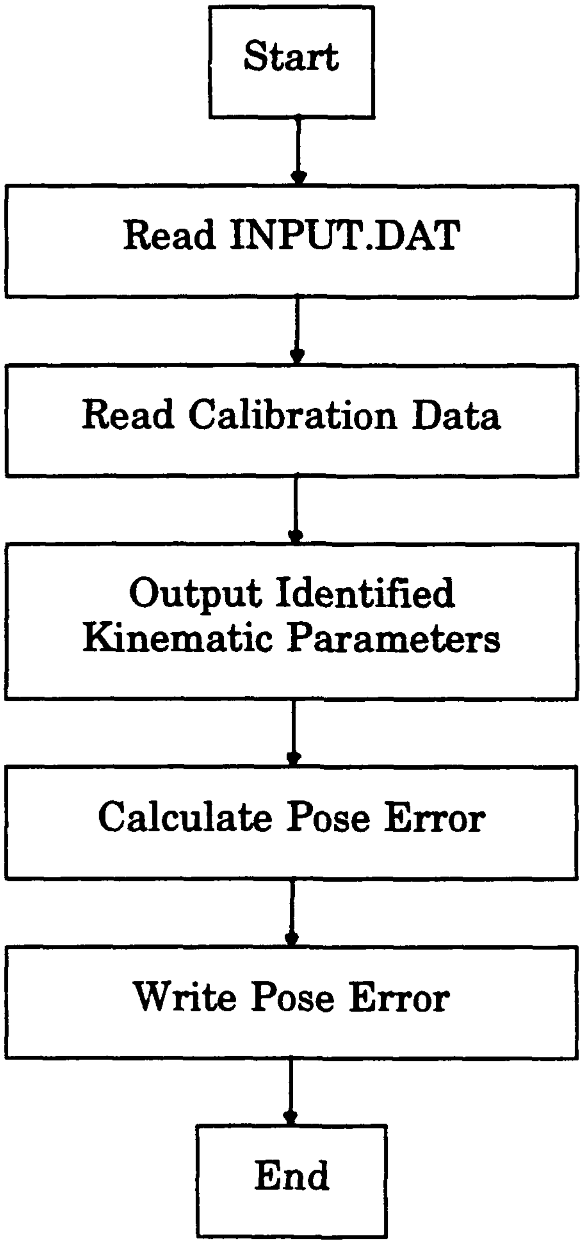


Figure 9. Flow Chart for Program ID6

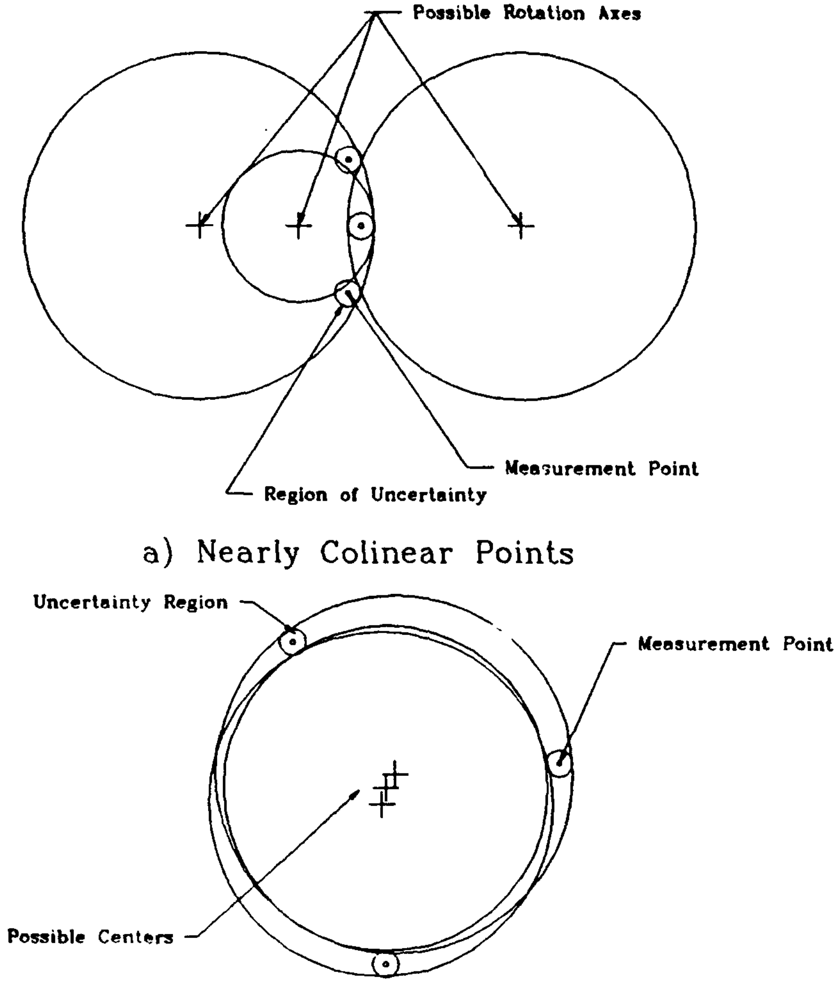
III. JOINT EXCURSION STUDY

A. BASIS

In this section the effect of total joint excursion during the conduct of a PUMA 560 manipulator calibration is studied. The null hypothesis of this study is that the greater the total joint excursion in each joint of the manipulator, the better the resulting calibration accuracy. Since calibration attempts to accurately identify the position and orientation of revolute joints, each joint must be rotated through a range that is large enough to ensure that the joint axis may be properly identified. Consider a joint rotated through only a small joint rotation. Any calibration point measured during this rotation will be nearly collinear. When measurement noise is considered a large variation in the identified axis of rotation is possible. When calibration data is obtained for a larger rotation of the joint the variation in the uncertainty of the position of the axis of rotation is much smaller. Figure (10) illustrates the effect of small versus large rotation on the identified axis of rotation. [Ref. 13]

B. METHOD

As previously discussed in Chapter I, calibration data from the PUMA 560 (see Appendix A) was analyzed to find the average position error using six poses selected from the data set to conduct the calibration. The resulting accuracy versus pose numbers selected were tabulated in Appendix B. As was discussed in Chapter I, these results



a) Nearly Colinear Points

b) Noncolinear Points

Figure 10. Effect of small vs. large joint revolutions on joint axis identification.

suggest that certain poses give better calibration than others. The study of this section uses the standard deviation of joint angles as a measure of the total joint excursion during each calibration experiment to see if there is a correlation between joint excursion and the resulting accuracy of calibration. It has been suggested [Ref. 14] that the accuracy of calibration is also a function of observation strategy and that for a fixed number of observations the best calibration is obtained when the observations are randomly distributed over the manipulator work ρace. Therefore, both joint space and work space end effector excursions were compared with the resulting position accuracy of the calibration. The correlation coefficient between position error and each of these variables was calculated to see if a model could be constructed which would allow the prediction of the calibration accuracy obtainable. The method of calculating the standard deviations and correlation coefficients is described below. The standard deviations of joint space and end effector excursion were calculated using program STAT1. STAT1 used the output file created by Swayze's program SCREEN, which consists of columns of position error with the pose numbers of the six poses used to calculate it, as an input file. STAT1 reads the position error and pose numbers from this file and then reads the joint angles and end effector positions from Swayze's calibration data files. The standard deviation of the joint excursion of each joint and the end effector excursion in x, y, and z is calculated and written to an output file along with the position error and the pose numbers. The output file was then input into MATLAB and the correlation coefficients between the position error and these standard deviations were calculated using MATLAB's CORRCOEF command.

C. RESULTS

Table 2 shows the correlation coefficients of the joint angle standard deviations for each joint and work space position standard deviation versus position accuracy.

Table 2. POSITION ERROR CORRELATION COEFFICIENTS.

Variable Compared	Correlation Coefficient
Joint 1	-0.3308
Joint 2	-0.6254
Joint 3	-0.0405
Joint 4	0.2163
Joint 5	0.0696
Joint 6	0.1774
X Variation	0.1540
Y Variation	0.0816
Z Variation	-0.2685

D. CONCLUSIONS

The extremely low correlation coefficients for joint angle excursion and work space position excursion versus. position accuracy shows that there is no correlation between how much joint angle excursion occurs during a calibration and the resulting accuracy of the calibration. The same conclusion can be drawn about work space position excursion. Thus an alternative hypothesis, that calibration accuracy is independent of joint space excursion and work space excursion seems to be proved. If optimum poses for calibration exist they must be found through other methods.

IV. CONDITION NUMBER STUDY

A. BASIS

In the manipulator calibration equation,

$$\delta \boldsymbol{P} = \boldsymbol{J} \delta \boldsymbol{K} \tag{29}$$

introduced in Chapter II, δP and J are not known exactly but have uncertainty. Measurement noise causes the uncertainty in δP and encoder noise causes the uncertainty in J [Ref. 15]. Consequently, there will be uncertainty in the solution for δK. If J is a square matrix, then

$$\frac{\|\delta K\|}{\|K + \delta K\|} < \kappa(J)\frac{\|\delta J\|}{\|J\|} \tag{30}$$

and

$$\frac{\|\delta K\|}{\|K + \delta K\|} < \kappa(J)\frac{\|\delta P\|}{\|P\|} \tag{31}$$

where

$$\kappa(J) = \|J\| \, \|J^{-1}\| \tag{32}$$

is the condition number of the identification Jacobian matrix. In general, the manipulator calibration equation will be an over determined system and J will not be a square matrix. In this case, the condition number will be given by:

$$\kappa(J) = \|(J^{T}J)^{-1}\| \, \|(J^{T}J)\|. \tag{33}$$

It can be seen from equations (31) and (32), that for a small deviation in either the identification Jacobian or the measured poses for calibration, coupled with a large condition number, will cause a large uncertainty in δK. That is, the accuracy of identification of the manipulator kinematic parameters will be degraded. Therefore it is hypothesized that the condition number of the identification Jacobian could be used as a predictor of the accuracy of the resultant calibration.

Driels and Pathre [Ref. 16] have shown that use of the condition number appears to be a good predictor of the expected relative resultant accuracy when comparing calibrations conducted with the same number of observations but with different observation strategies. They achieved the same results using the condition number to differentiate between calibrations using the same observation strategy but with different numbers of observations. In this study, the utility of the condition number, as a predictor of calibration accuracy when comparing calibration experiments using the same observation strategy and number of observations, was studied. That is, can the condition number of the identification Jacobian be used to predict the best poses to use for calibration?

B. METHOD

The computer program TEST used by Swayze to calculate the position error from calibration data was modified to also calculate the L1 condition number for each set of poses tested. The program was run for 80 different sets of poses. The resulting accuracies and condition numbers were then compared. The poses selected for this study were in four groups as follows:

- The twenty sets of poses which gave the lowest position error. The position error ranged from 0.464 to 0.619 mm.

- Twenty sets of poses which gave a position error in the range of 0.988 to 1.000 mm.

- Twenty sets of poses which gave a position error in the range of 20.15 to 21.64 mm.

- The twenty sets of poses which resulted in the largest position error of any sets considered. The position error ranged from 48.52 to 198.94 mm.

C. RESULTS

All of the position error versus condition numbers, along with the pose numbers used in the calibration, are tabulated in Appendix (B) and are summarized in Table (3).

Table 3. POSITION ERROR VS. CONDITION NUMBER

	Range of	
Group	Position Error (mm)	Condition Number(x10^6)
1	0.464 - 0.619	0.095 - 1.8
2	0.988 - 1.000	0.12 - 1.2
3	20.15 - 21.64	3.2 - 125.4
4	48.52 - 198.94	14.9 - >1000

D. CONCLUSIONS

Although the general trend of the condition number versus position error is as expected, (i.e. higher position error equates to higher condition number) the variability in condition number within a group of data is large enough to prevent any accurate predictions of the calibration accuracy given only the condition number.

V. ADS SEARCH FOR OPTIMUM POSES

A. BASIS

In Chapter I the motivation for a search for optimum poses for calibration was presented. In Chapters III and IV the condition number and joint excursion studies failed to detect any special significance between small sets of poses that produced superior calibration results and the sets which achieved poor calibration results. Clearly, if optimum poses exist, and they appear to, they must be found through other means. The hypothesis for the study of this chapter is that a numerical optimizer should be able to find the best set of poses to calibrate the PUMA 560 manipulator if an adequate objective function can be found. Since the goal in using an optimizer is to improve calibration accuracy it would seem reasonable to use position error as an objective function to be minimized. Since the absolute position error can never be known in a real manipulator, computer simulation studies were conducted so that its absolute position and orientation error could be known.

B. METHOD

A computer program, OPT6A, was written to conduct simulated calibration experiments. Figure (11) is a logical flow chart that shows how OPT6A works.r At the core of the program is a numerical optimizer, ADS, which is given an initial guess of the best sets of joint angles for conducting a calibration. In this simulation the exact

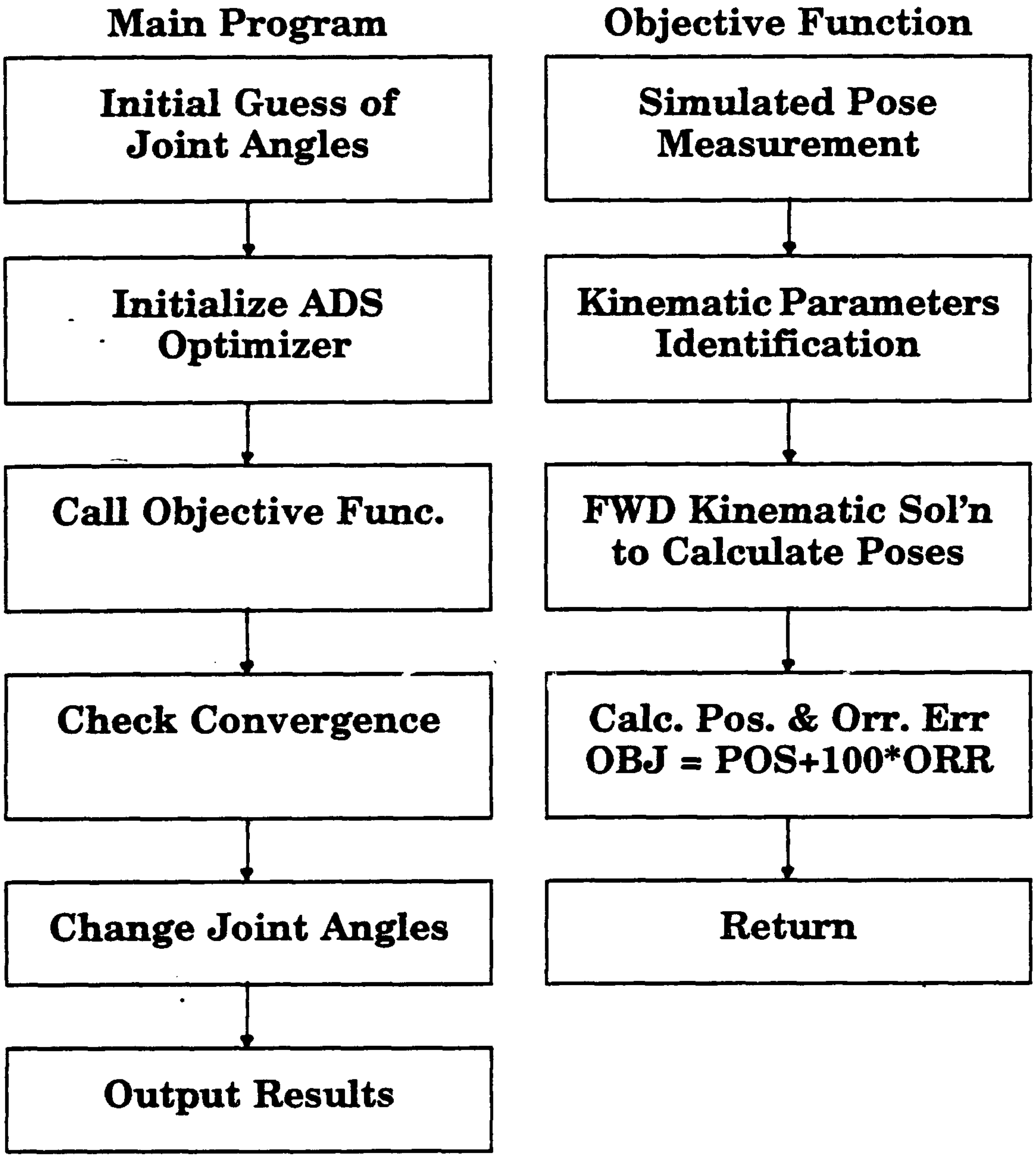


Figure 11. Flow Chart for Program OPT6A

kinematic parameters for the manipulator to be calibrated are known. If the optimizer can find a set of joint angles which allows the exact kinematic parameters to be identified by the parameter identification algorithm, then this would be an optimum calibration. In each iteration of the optimizer the program calculates an objective function by a series of steps.

- Simulated Pose Measurement Step: The program POSE [Ref. 17] is used to calculate the poses (associated with the joint angles) which would have been measured in the laboratory if this were a real vice simulated experiment. The poses produced have random noise injected to model the measurement process.

- Parameter Identification Step: The poses produced above are used by the parameter identification algorithm similar to program ID6. The nominal kinematic parameters, with which the algorithm starts, are different by a known amount of 10 mm. for lengths and 5 degrees for angles. This ensures that the algorithm has a non-trivial identification problem.

- Comparison Step: The program POSE is used again with the kinematic parameters identified in the previous step to find the forward kinematic solution of the joint angles in the verification data set. This set contains a relatively large number of sets of joint angles and poses that were produced with the exact kinematics and no measurement noise. The difference between the verification data set exact poses and calculated poses is then computed. The RMS position and orientation errors are then calculated.

- Objective Function Construction: The orientation error has typically been found to be about 100 times smaller than the position error when positions are expressed in millimeters and angles are expressed in degrees. Therefore, the objective function used is the position error plus 100 times the orientation error.

ADS then calculates the gradient of the objective function numerically and changes the joint angles in a direction which will reduce the objective function. This process repeats until ADS can find no further improvement in the objective function. The program then terminates and writes the joint angles and identified kinematic parameters to a file.

C. RESULTS

The program was first tested with no simulated measurement noise. ADS conducted 72 objective function calls and converged to the initial guess joint angle set. ADS takes two function calls per design variable to compute the gradient. The program was using 6 poses with 6 joint angles each for a total of 36 design variables. Thus ADS converged to a solution which identified the exact kinematic parameters in the minimum number of function calls to achieve convergence. This was repeated several times with different set of randomly selected starting joint angles. The result was always the same, convergence in 72 function calls without changing the starting joint angles.

The program was next run with simulated measurement noise. In this case the program did not converge. Due to the randomly applied simulated measurement noise the parameter identification algorithm did not identify the exact kinematic parameters. This caused a position and orientation error during each objective function call. Therefore, the objective function gradient is now non-zero and ADS changes the joint angles. With the new joint angles random noise is again applied in the simulated measurement step. The gradient calculated with these joint angles is also non-zero. After every gradient calculation (each 72 objective function calls) ADS changed the joint angles but they always stayed very close to the starting joint angles. This result was seen even if the starting point was a set of joint angles identified in Swayze's work as giving a poor calibration.

D. CONCLUSIONS

In the case of zero measurement noise the identification algorithm converges to the exact kinematic parameters on the first calculation of the gradient and this makes the gradient zero regardless of any dependency that the joint angles may have on resulting calibration accuracy. In the case of randomly injected simulation noise the identification algorithm could not identify the exact kinematic parameters and thus, the gradient can never be zero. However, the gradient due to the noise is small but large enough to keep the optimizer from converging. Therefore, if there is a dependency of calibration accuracy on joint angles (and hence the poses), the optimizer will tend to move the selected joint angles toward the optimum pose. No such movement of joint angles was detected so the conclusion must be that if such a dependency exists, it is only a very weak function of joint angles which produces such a small gradient that even the small simulated measurement noise completely obscures its effect.

E. RETEST OF BEST POSES

With the negative results of three totally separate investigations, serious doubts arise about the validity of the starting premise of this work, that is, that there exists select small sets of poses which will give nearly the same calibration accuracy as a large set of randomly picked poses. The research conducted thus far has found no evidence that this premise is true. Why then, did Swayze find such sets?

In an attempt to answer this question an attempt was made to duplicate his results. Using the same PUMA 560 manipulator and the same coordinate measuring machine

(CMM) that Swayze used, the six poses previously found to give the smallest position error of any six poses were measured again. These were the poses identified in his data as pose numbers 8, 11, 17, 18, 19 and 20. The resultant six poses were input as the experimental group in program TEST with the resulting position error of 4.48 mm when compared to Swayze's control group of poses. This error is approximately 100 times greater than the error of 0.464 mm originally obtained for these poses. The conclusion to be drawn from this result is that measurement error, though small, has a very big impact on the positional accuracy of the resultant calibration.

F. POSSIBLE ALTERNATE HYPOTHESIS

The results obtained in the previous section leads to the conclusion that the null hypothesis of this work must be rejected. A possible alternate hypothesis based on the no noise case described above is now proposed as follows. The positional accuracy of a manipulator which is achieved via calibration is a function of only the measurement noise in the calibration data and is totally independent of the joint angles used. This could explain why Swayze was able to find a small number of subsets of his calibration data which were able to give very good positional accuracy. The poses in these small subsets would be the ones which, for whatever reason, have smaller measurement error than the average pose. The poses in the small subsets which produced poor positional accuracy would be the ones which have a relatively larger measurement error.

VI. ZXSSQ SEARCH FOR OPTIMUM POSES

A. BASIS

In the last chapter, the program OPT6A which uses the optimizer ADS was presented. In this chapter, a similar program, OPT6C, is presented. OPT6C was written to use the IMSL subroutine ZXSSQ as the core subroutine vice the public domain program ADS. This was done for two reasons. First, there is technical support available for ZXSSQ while the user of ADS is not well supported. The second reason for rewriting OPT6A to utilize ZXSSQ in place of ADS is that OPT6A did not converge in the case where simulated measurement noise was injected into the calibration data. If the second optimizer, which uses completely different source code, also failed to find an optimum pose it would add to the body of evidence against the null hypothesis of this thesis.

B. METHOD

Figure (12) is a flow chart for OPT6C. The program uses two applications of ZXSSQ. The first application of ZXSSQ was renamed ZXSSQ1 to prevent fatal interaction between the two applications during program execution. The first application of ZXSSQ selects the joint angles for which a simulated calibration experiment is run. The selection is based upon trying to minimize the objective function. The objective function is calculated using the same four steps as previously described in Chapter V for program OPT6A briefly described as follows:

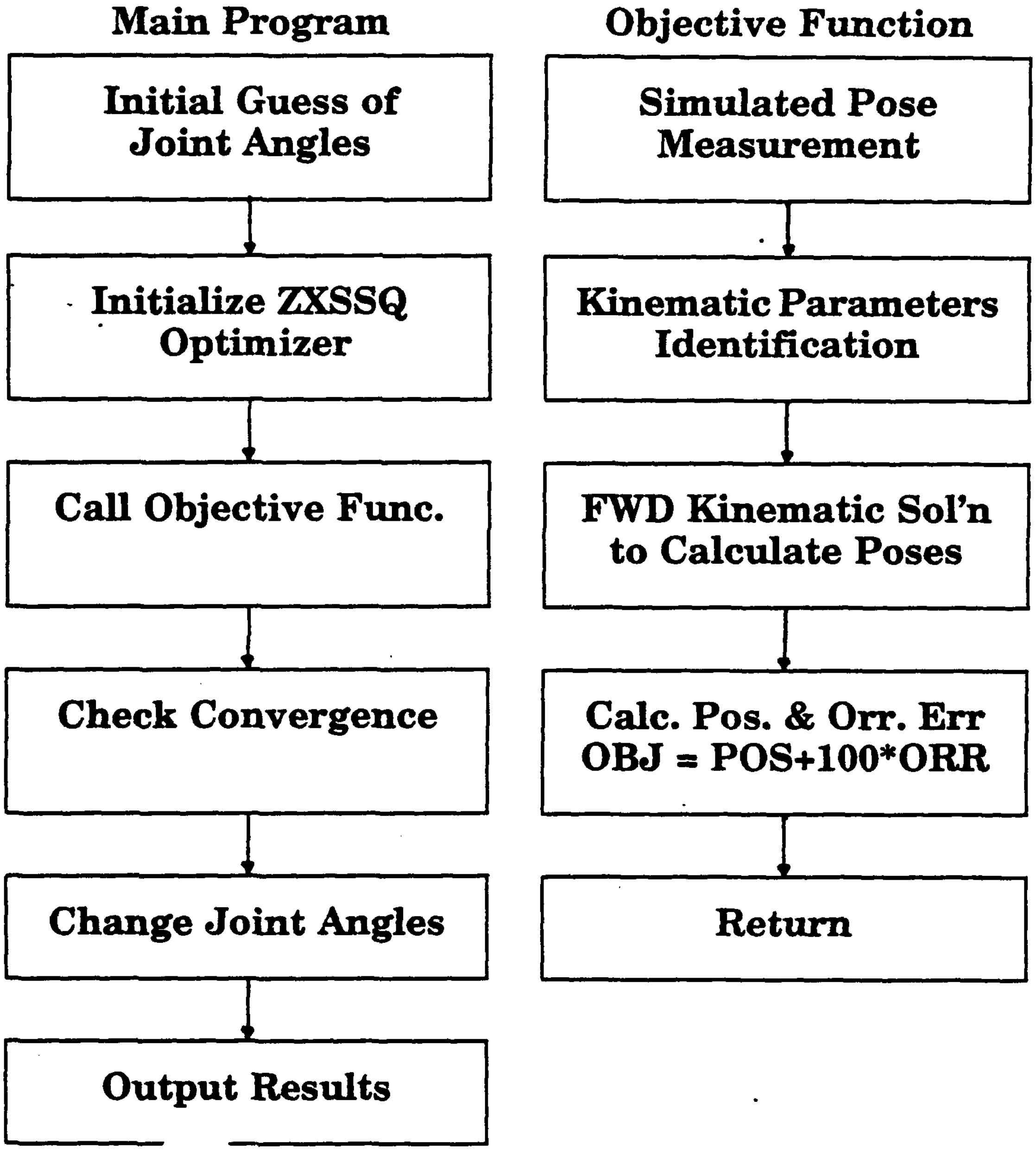

Main Program
Initial Guess of Joint Angles
Initialize ZXSSQ Optimizer
Call Objective Func.
Check Convergence
Change Joint Angles
Output Results
Objective Function
Simulated Pose Measurement
Kinematic Parameters Identification
FWD Kinematic Sol'n to Calculate Poses
Calc. Pos. & Orr. Err OBJ = POS+100*ORR
Return

Figure 12. Flow Chart for Program OPT6C

- Simulated Pose Measurement
- Parameter Identification: This step uses the other application of ZXSSQ in an ID6 type program
- Verification Step: Calculates position and orientation RMS errors
- Calculate the objective function and pass it to ZXSSQ1.

Based on the results of the simulated calibration experiment ZXSSQ1 changed the values of the joint angles and ran another simulated calibration experiment. When the objection for function for ZXSSQ1 does not change the program has converged to an optimum set of joint angles for calibration.

C. RESULTS

The results from OPT6C were exactly the same as for OPT6A for both the no measurement noise and measurement noise added cases. In the no simulated measurement noise case the outer loop optimizer, ZXSSQ1, converged after enough function calls to calculate the gradient to the starting joint angle set. When simulated measurement noise is added the program does not converge but does change the joint angles after each gradient evaluation is complete. If a gradient due to a dependence on joint angles were present the joint angles selected would tend to change in the direction of the optimum set of joint angles. Just as in program OPT6A no trend in the movement of the joint angles is observable and the joint angle set stays essentially the same as the starting set.

D. CONCLUSIONS

In the no noise case the optimizer converges to the starting joint angle set independent of the starting joint angles independent of which optimizer is used to select the joint angles. In the case where simulated measurement noise was added to the calibration data the noise affected the gradient computed at each step only a small amount but even this small amount totally obscured any dependence of positional accuracy on joint angles used. That is, the gradient due to the measurement noise is superimposed on an essentially flat topology.

VII. RANDOM SEARCH FOR OPTIMUM POSES

A. BASIS

In the previous two chapters two numerical optimizers were shown to have been unable to detect a dependency of the objective function on joint angles. Two possible reasons for this are apparent. First, there may be no dependency to be found which means that there are no optimum poses to be found, or second, the dependency may be a very weak function of joint angles which the optimizers failed to detect because they have tested a neighborhood around the starting point which was too small. The purpose of the investigation of this section was to check large neighborhoods for objective function gradients. If an optimum set of poses exists it will result in an objective function at that point which has a lower value than other point.

B. METHOD

In this chapter the method of search for an optimum pose was changed from that of the previous two chapters. Instead of changing the joint angles based on the objective function, as was done in both programs OPT6A and OPT6C, in this search random points were picked in a neighborhood around the starting point. At each point selected the objective function is calculated. Each time the objective function is calculated the simulated measurement noise injected in the calibration poses is different because it is randomly selected. To get a measure of the range of calibration error resulting from this noise the objective function is calculated 100 times at each randomly selected point in the neighborhood of the starting point. The mean and standard deviation of these 100 objective function calculations is computed and written to a file. This process is repeated for different sized neighborhoods ranging from ±5 to ±180 degrees of variation from the starting angle for each of the six joints of the PUMA 560 manipulator. The objective function used is the same as the objective function used in OPT6A and OPT6C, position error plus 100 times orientation error. All of the data obtained for this chapter was obtained using a version of the program OPT6C that was modified slightly to defeat the joint angle selection subroutine ZXSSQ1 and call the objective function directly vice let it be called from ZXSSQ1. Each of the neighborhoods tested was tested with 100 random points. The 180 degree variability case was run ten times to give a total of 1000 random points tested throughout the working volume of the manipulator.

C. RESULTS

Each of the 20 random search data runs conducted used the same set of six poses and associated joint angles as the starting point (center of the neighborhood). The test results are presented in Table (4). The objective function of the starting point was calculated 100 times to give a comparison for each of the other points tested. The degrees column represents the magnitude of the maximum variation in each joint angle during the test. Each row in the table (except the 0 degree row which is neighborhood consisting of only one point) represents the low and high of 100 data points in the specified neighborhood of the starting point. The low and high refer to the

Table 4. OBJECTIVE FUNCTION DATA SUMMARY

	Low Data		High Data	
Degrees	Mean	Std Dev	Mean	Std Dev
0	5.46	1.45	NA	NA
5	5.07	0.99	6.62	2.15
10	4.71	1.14	8.16	2.79
20	4.66	1.06	13.28 186.41*	7.15 5.38*
30	4.78	1.03	22.63	15.50
40	4.30	0.83	23.76	14.68
60	4.78	1.25	19.28	12.30
80	4.39	0.90	38.83	24.22
120	4.75	1.08	25.84	13.28
160	4.71	1.03	17.44	9.15
180-1	4.18	0.98	21.74	13.78
180-2	4.62	0.91	32.54	17.71
180-3	4.74	0.99	53.43	27.47
180-4	7.69	2.53	10.17	4.35
180-5	4.00	0.75	17.17	9.46
180-6	4.04	0.97	26.65	21.54
180-7	4.47	1.10	27.77	47.01
180-8	4.84	1.05	19.66	7.97
180-9	4.38	0.87	28.87	18.24
180-10	5.20	1.38	28.13	17.60

* The 186.41 mean was the highest with 13.28 next highest. It appears that this point was very near a singular point.

lowest/highest mean of the 100 points tested for that particular neighborhood. In each of the runs presented, the measurement error injected was 0.2 mm for position and 0.2 degrees for angles times a random number between -1.0 and 1.0.

D. CONCLUSIONS

A review of the means and standard deviations of the low data shows that the mean objective function does not change appreciably throughout the working volume of the manipulator. This means that there is no optimum set of six poses (and associated six sets of joint angles) which will give a calibration accuracy that is as good as a large set of calibration data. That is, if the calibration data all contain a random measurement error there can be no guarantee of a good calibration unless a large number of poses is used to calibrate the manipulator. A review of the high data shows that there are, however, sets of poses which produce relative poor calibration results. These sets are those which in some way approach a singularity of the manipulator.

VIII. DISCUSSION, CONCLUSIONS AND RECOMMENDATIONS

The null hypothesis of this work was that there exists select small subsets of poses which will give calibration accuracy approaching the repeatability of the manipulator. Previous experimental work conducted by Swayze had shown that there were such small subsets. The goal of this thesis was to investigate these subsets and discover what is unique about them, how to find them and then use them to predict calibration accuracy prior to actually conducting the calibration. In pursuit of this goal a three pronged investigation of the problem was conducted as follows:

- Investigate the condition number of the calibration Jacobian matrix.
- Conduct statistical analysis of sets of poses which yielded good and poor calibration results.
- Conduct computer simulation studies searching for optimum poses.

During these investigations several interesting discoveries were made:

- The condition number of the calibration Jacobian matrix is not an effective predictor of calibration accuracy.
- There is no statistical difference between either the joint space excursions or work space position excursions of subsets of calibrations which produced good calibration accuracy and those which produced poor calibration accuracy.
- The computer simulation studies could not find a set of poses which produced a better calibration than a similar number of poses chosen at random.
- One set of poses (obtained by Swayze) which were known to give a calibration accuracy approaching the repeatability of the PUMA 560 manipulator gave a much poorer result when they were measured again.

Based on these discoveries it is concluded that ᴗptimum poses for conducting kinematic parameters identification do not exist. Thus, the following alternate hypothesis appears to be proven. The accuracy which results from conducting a manipulator calibration is a function of the pose measurement error only. It has been shown that without measurement noise any set of poses used during calibration will yield the exact kinematic parameters.

A recommendation for further study is to investigate an optimum way to overcome the affects of measurement error. Given that there will be measurement noise:

- Is the measurement noise randomly distributed?
- Does the measurement noise also have a bias?
- Can a method be discovered to tell which poses have the most noise?

APPENDIX A

The following 42 data sets each consist of six joint angles and a four by four transformation matrix. The last row of the transformation matrix is [0 0 0 1] and is inserted by the computer program which analyzed the data, TEST.FOR. This data was obtained by Swayze [Ref. 18] and is published here only because it has not previously been published and the data has been used in the research for this thesis. The numbers in parenthesis are the pose numbers used Swayze.

Experimental Group of Poses from file 121.dat:

```
(1)
 108.8970        39.33100        39.12200        169.9640        -99.93700        155.5300
 0.7711439       0.6178218      -0.1544949        752.0477
-0.2010305       5.1664468E-03 -0.9781566         178.0240
-0.6032777       0,7854300       0.1283638        302.2741
(2)
 100.3600        42.98400        46.86200       -33.97000        -85.36900        203.5500
-0.7042692       0.4168146       0.5771767        743.7152
 0.4851753      -0.3148709       0.8149259        311.4646
 0.5205824       0.8528200       2.1506701E-02    330.3344
(3)
-109.6930        137.0600       -170.7660        3.812000        -90.80800        265.9100
-0.1518535      -0.9553381      -0.2543225         23.16502
 0.8186871       2.1206835E-02 -0.5741130         433.8619
 0.5535835      -0.2953570       0.7792214        205.5811
(4)
-116.2900        142.2890        155.7200       -59.72700         79.77200        187.0300
 0.3266630       0.4924835       0.8074195         37.15068
-0.8335165      -0.2516308       0.4923119        435.8255
 0.4461260      -0.8334738       0.3281367        238.0869
(5)
-38.60600        137.0000        153.6440        99.16800          99.95400        192.2500
-0.4642738      -0.4837569      -0.7427612        768.0645
 0.8433744      -0.4885970      -0.2123972        260.5175
-0.2607758      -0.7246256       0.6346267        279.1954
(6)
-50.18600        143.4540        171.2220       -5.806000         -93.86200        265.7900
 0.51136105E+00  -0.80244593E+00  0.31021404E+00  0.74899583E+03
 0.64424440E+00   0.11321671E+00 -0.75616163E+00  0.38264032E+03
 0.57162754E+00   0.58653736E+00  0.57488381E+00  0.31173157E+03
(7)
 81.76600        42.98400       -11.20100        8.707000          99.94300        7.890000
 0.9714892      -0.2401149       7.1078627E-03   463.9761
 0.1805270       0.7110476      -0.6789809        465.9579
 0.1587303       0.6609058       0.7330764        249.1465
(8)
 93.77900        36.86500        23.62600       -109.1820          99.36000        59.88000
-0.2977736       0.2052929      -0.9341167        457.6155
 8.6175419E-02  0.9789167       0.1878993        471.2484
 0.9512731      -2.6878638E-02 -0.3097883        273.7163
```

```
(9)
 78.10200       42.98400       21.11600        13.93100        99.92600       77.04000
 2.6521383E-02 -0.9871532      0.1542757        409.0047
 0.2885189     -0.1427143     -0.9467071        260.9548
 0.9565622      7.0415117E-02  0.2812690        358.4876
(10)
 79.30500      -77.64600      -146.8490        18.36900        51.22400       219.7700
-0.4150131      0.8727108      0.2576438        495.5172
-4.9237855E-02  0.2600399     -0.9634876        474.5608
-0.9074232     -0.4129355     -6.4918131E-02    43.72158
(11)
 81.71600      -58.92000      -169.6070        91.46700       -4.532000       265.8700
 0.9838807     -6.5087982E-02 -0.1725709        432.4085
-6.3742168E-02  0.7584915     -0.6490363        485.6690
 0.1730159      0.6495744      0.7405714        294.8966
(12)
 97.02600      -23.85100      -149.4960       -87.84700        71.47700      -60.72000
-0.2688300      7.3175065E-02 -0.9597657        460.3569
 0.8703721     -0.3997912     -0.2770904        175.6644
-0.4051149     -0.9121282      4.0358867E-02   -24.91388
(13)
 72.82800      -6.295000       163.2130        101.6240        75.67900      -149.4300
 0.5603818      2.2149209E-02  0.8270285        454.9760
 0.1467601      0.9801015     -0.1238874        111.6274
-0.8144436      0.1901652      0.5454870        322.2652
(14)
-89.78600       137.0160      -170.7720       -109.9290        80.29400      -124.9700
-0.1007417      0.3816238      0.9202351        365.3622
-0.9943419     -1.1903070E-03 -0.1072020        493.6407
-3.8403843E-02 -0.9243201      0.3796740        242.3944
(15)
-83.62800       137.0160       154.1820       -4.466000        82.35400      -94.99000
-8.2023442E-03 -0.9976611      7.3819250E-02    328.9090
-0.2322559      7.3571846E-02  0.9678066        419.2001
-0.9701620     -1.0517023E-02 -0.2323161        317.8822
(16)
-57.29400       137.0160       119.1410        85.65500        99.97000      -78.07000
-0.3508146     -0.1433183     -0.9254565        529.9622
 3.8676374E-02 -0.9887519      0.1358849        48.67244
-0.9345216      1.2880162E-02  0.3530353        351.0218
(17)
-71.29600      -107.0340      -32.52000        82.55700        3.834000       116.4100
 0.99062566E+00  -0.10410171E+00   0.86026423E-01   0.38839199E+03
 0.13228540E+00   0.61397810E+00  -0.77563480E+00   0.42577865E+03
 0.29880215E-01   0.78071063E+00   0.62298021E+00   0.11054153E+03
(18)
-75.42100      -128.0350       17.91900        2.280000       -70.73000       58.72000
-0.67965640E+00   0.73258194E+00   0.20650149E-01   0.41289990E+03
 0.38539392E-02   0.33905209E-01  -0.10011758E+01   0.48829670E+03
-0.73514193E+00  -0.67983649E+00  -0.26328799E-01   0.29261114E+03
(19)
-56.49700      -155.5170      -19.55000       -79.63400       -32.82700       218.3100
 0.22061610E+00   0.86723512E+00  -0.44271410E+00   0.43055592E+03
-0.34748249E+00  -0.35431225E+00  -0.86729930E+00   0.93410809E+02
-0.91028359E+00   0.34490190E+00   0.22218084E+00   0.56289492E+02
(20)
-48.28500       137.0540       126.1450       -80.89800       -99.99800       217.4500
 0.34238403E+00   0.40383149E+00  -0.84669893E+00   0.65488575E+03
-0.87378807E+00   0.47442192E+00  -0.12505314E+00   0.64530311E+02
 0.35034718E+00   0.78282495E+00   0.51597404E+00   0.33559238E+03
(21)
 105.6940       42.96200       64.74200       -81.89800        84.54500       237.3700
-0.27501634E+00  -0.10263554E+00  -0.95698159E+00   0.60294286E+03
 0.73135899E+00  -0.66452849E+00  -0.14118894E+00   0.51142455E+02
-0.62183336E+00  -0.73956944E+00   0.25659018E+00   0.32238865E+03
```

Control Group of Poses from file 221.dat

```
  35.93600      -35.34300      -169.4750       18.23700        35.92000        44.14000
 -0.42128912E+00  -0.87290439E+00   0.24938342E+00   0.59045937E+02
 -0.79106325E-01  -0.24037452E+00  -0.96867055E+00   0.21109254E+03
  0.90376931E+00  -0.42828457E+00   0.33492941E-01   0.54996885E+02
 48.91700       -40.15500      -176.7100       27.08100       39.45700       -4.130000
 0.46992486E+00  -0.85767772E+00  0.21442382E+00  0.12791065E+03
 0.26520422E+00  -0.99985722E-01 -0.95832837E+00  0.29080905E+03
 0.84452851E+00   0.50493489E+00  0.18185119E+00  0.18761016E+03

 44.72000       -45.16500      -170.3490       61.38100       22.98300       10.16000
-0.3544885      -0.9348714     -2.3160055E-02   81.27153
 0.4876479      -0.1653767     -0.8582276       326.1524
 0.7977632      -0.3159772      0.5148042       185.8410

 53.77300       -43.78100      -170.3490       50.85000       22.98300       -9.060000
 0.2627726      -0.9638667      6.7367002E-02   177.3687
 0.4302469       4.8305068E-02 -0.9004627       305.4116
 0.8667043       0.2642176      0.4267416       217.1807

  66.47300       -34.82100      -166.6130       59.38100      -60.29800       -9.070000
 0.1917356      -0.3510893     -0.9180831       188.4730
 0.1960170       0.9285316     -0.3155827       280.9062
 0.9625344      -0.1210259      0.2460115       195.6472

  72.26300       -23.54400      -173.9630       47.15900      -53.13000       34.98000
 6.5060005E-02 -0.6402318     -0.7648578       259.8408
 0.6941696       0.5781585     -0.4281123       180.1360
 0.7165292      -0.5032795      0.4816219       246.3320

  86.03400       -24.79600      -173.6010       62.51200      -56.09100       13.58000
 0.2489289      -0.5986599     -0.7626784       397.7957
 0.4983466       0.7524565     -0.4297198       190.9869
 0.8305680      -0.2739021      0.4854273       243.0207

  69.00500       -30.98700      -151.5620       37.44100      -63.85800       109.0800
-0.53423728E+00  -0.35270538E+00  -0.77019241E+00   0.27880969E+03
 0.83560081E+00  -0.33817895E+00  -0.43064067E+00   0.20406729E+03
-0.11112888E+00  -0.87301113E+00   0.47345155E+00   0.11770727E+03

  116.3340       -26.23500      -176.4350       94.41700      -65.58800       36.98000
-0.98605457E-01  -0.65058529E+00  -0.75487258E+00   0.68320940E+03
 0.81147807E+00   0.38417210E+00  -0.43868160E+00   0.21458974E+03
 0.57550064E+00  -0.65589635E+00   0.48987606E+00   0.14080517E+03

  36.20000       -27.98200      -148.5240       3.812000      -69.37300       62.12000
-0.77480661E-01  -0.63090727E+00  -0.77416902E+00   0.68819906E+02
 0.79874436E+00   0.42335251E+00  -0.42742825E+00   0.18357909E+03
 0.59753727E+00  -0.65146330E+00   0.47102512E+00   0.13537724E+02

  30.482         -27.993         -169.4920       0.088          15.985         28.7
  0.13400568e-02 -0.10008975e+01 -0.17553490E-01 0.12662462E+01
 -0.16022098E-01 0.16143588E-01 -0.10015072E+01 0.13709585E+03
  0.99987859E+00  -0.55218105E-03 -0.15364223E-01 -0.48795680E+00

  46.12100       -38.83100      -164.3830       6.669000      -95.74000       21.03000
 0.7159228       7.9949401E-02 -0.6951312       36.17091
 0.3594961       0.8098706      0.4646400       416.0708
 0.6000932      -0.5825518      0.5510759       165.2429

  55.82700       -46.14800       168.4810       157.2140      -20.00600       70.41000
-0.1752464       0.8494951     -0.4954682       63.80724
-0.4439764      -0.5176451     -0.7323478       474.5010
-0.8792164       9.2117548E-02  0.4667683       315.4322

  103.4250       -18.63800      -165.7230       124.4640       33.34900       133.3300
-0.2645359       0.7867539      0.5574042       721.4144
-0.4907337       0.3859897     -0.7797947       75.30141
-0.8286433      -0.4802224      0.2835915       78.12855

  111.5550       -4.307000       144.2230      -66.14900       28.47700       221.2900
-0.7327883      -0.6123065     -0.2979847       755.8734
 0.5532924      -0.2809739     -0.7848600       84.49096
 0.3970938      -0.7398301      0.5446114       319.5979

 -115.8950       -120.0530      -27.55900      -6.092000       55.78300       265.9000
 9.4917268E-02 -0.9250252     -0.3678622       28.97224
 0.9904016       0.1182010     -4.0311139E-02   452.6450
 8.0429345E-02 -0.3591460      0.9283856       44.94406
```

```
  -104.1340      -130.0230       14.01300        6.119000      -44.92300      -28.15000
 -0.9903060     -0.1423901     -3.0212890E-02   80.85344
 7.4401505E-02  -0.3172578     -0.9462133       447.9015
  0.1256082     -0.9385339      0.3250673       248.8567

  -39.46800      -112.7800      -25.03800       16.08400       14.71100       174.7800
  0.7682810      0.2776003      0.5773852       714.7224
  0.1234613      0.8218290     -0.5548360       447.9086
 -0.6287887      0.4974241      0.5969992       92.48531

  -42.29200      -144.8820       24.78000      -80.75500      -99.33300       246.1000
  0.3684531      0.4557302     -0.8084836       708.3550
 -0.8322740      0.5551271     -6.5130509E-02   454.5897
  0.4186503      0.6972162      0.5839717       291.9995

   60.32000       42.94600       10.00900      -76.11900       77.44800       177.9100
  0.1196614      0.2375790     -0.9647583       69.73510
  0.7734914     -0.6277151     -5.9824646E-02   449.8174
 -0.6206928     -0.7408679     -0.2615671       206.8124

   55.90400       39.78700       10.00900       166.1240      -68.76900       244.2500
  0.1356188     -0.6933872     -0.7097770       57.00895
  0.7253505      0.5556322     -0.4053136       438.8703
  0.6750536     -0.4601158      0.5784187       301.4963
```

APPENDIX B

The following data is from two sources. The position error was generated by Swayze using the program TEST. The six pose numbers (which represent the poses from Appendix A) yielded a calibration with position errors shown. The L1 condition numbers were generated using a modified version of TEST which was rerun using the same calibration data.

Group 1 Data:

Position Error	Pose Numbers Used						L1 Condition Number
0.464226	8	11	17	18	19	20	489388.
0.491716	3	8	9	11	18	19	1152313.
0.526177	2	9	11	17	18	20	1103019.
0.512118	2	9	11	13	17	20	610886.
0.550705	3	6	8	11	17	18	94999.
0.577800	4	6	11	17	18	19	312798.
0.580232	5	9	11	17	18	20	1798603.
0.581695	3	5	8	10	11	18	1207517.
0.583514	2	3	5	7	9	11	542206.
0.584908	1	9	13	18	20	21	1041269.
0.584987	2	5	9	11	18	20	512458.
0.585944	2	9	11	12	17	18	557937.
0.606438	9	13	15	18	20	21	582533.
0.608049	2	4	7	9	11	17	265467.
0.608943	9	14	17	18	19	20	790714.
0.614340	3	6	8	17	18	19	961907.
0.615247	4	7	10	16	17	20	902923.
0.622127	3	6	7	9	17	19	235194.
0.622639	5	9	15	17	18	20	245892.
0.619353	1	8	9	15	17	20	744273.

Group 2 Data:

Position Error	Pose Numbers Used						L1 Condition Number
0.988286	5	6	9	12	13	19	395734.
0.988995	1	3	4	11	17	19	349937.
0.989457	2	3	4	10	11	17	1042771.
0.989890	7	9	12	15	19	21	320270.
0.991495	1	3	6	8	13	17	122501.
0.992349	5	8	11	14	18	19	486052.
0.992768	6	11	15	16	18	19	709844.
0.993289	2	6	10	12	19	21	289391.
0.993329	7	8	9	15	17	19	1025013.
0.993746	7	12	14	17	19	20	124090.
0.993841	6	9	17	18	19	21	1152692.
0.994538	2	4	9	10	17	21	578332.
0.995158	1	4	7	9	11	12	716899.

Group 2 Data continued:

Position							L1 Condition
Error	Pose Numbers Used						Number
0.995230	6	9	11	12	17	19	591913.
0.997206	6	7	8	11	16	19	380416.
0.997569	4	10	11	16	17	19	136973.
0.998109	7	8	11	14	16	20	940426.
0.980157	2	9	10	15	19	21	420084.
0.998841	3	4	6	12	13	17	205866.
0.999507	2	4	5	8	19	20	118092.

Group 3 Data:

Position							L1 Condition
Error	Pose Numbers Used						Number
20.146641	3	5	6	9	15	16	73198110.
20.147523	3	4	6	10	19	21	14063011.
20.294230	4	5	7	13	14	21	4688634.
20.405239	3	9	15	16	20	21	11522027.
20.439629	5	7	8	9	12	16	47877894.
20.471388	5	8	10	11	12	16	36365066.
20.650280	3	4	5	6	16	19	9754510.
20.656707	7	12	13	14	16	17	3221401.
20.718560	1	9	10	11	12	21	65400878.
20.964527	1	2	5	15	16	21	9115647.
21.014691	2	5	14	16	20	21	19179810.
21.129651	1	2	8	10	11	12	80165079.
21.159717	2	4	5	14	16	17	125357914.
21.333181	1	2	4	15	16	21	10910522.
21.387209	2	3	6	11	13	17	16536973.
21.494541	1	2	5	6	9	15	23163331.
21.497572	1	3	7	9	14	21	11560094.
21.535572	1	2	8	9	19	21	4146237.
21.569552	4	5	8	9	15	16	7292038.
21.642854	3	4	6	11	13	19	8376847.

Group 4 Data:

Position							L1 Condition
Error	Pose Numbers Used						Number
48.523596	2	3	6	7	15	16	95704374.
50.152168	1	2	7	14	15	21	283677767.
51.076583	1	2	3	4	5	8	14919023.
52.333366	1	2	4	13	16	18	82284953.
52.818301	1	2	9	14	16	21	104471277.
53.212275	2	5	7	8	9	16	74515433.
60.366057	1	2	3	5	6	8	41631096.
61.222631	1	2	4	5	15	16	172282800.
61.840316	5	8	11	12	16	19	152144311.
63.005565	1	3	5	7	16	21	21494095.
63.460522	1	2	13	15	16	19	202197988.
160.713551	5	8	12	12	13	16	**********
63.507128	5	8	9	12	13	16	87466901.
66.215422	1	2	7	8	9	21	139893825.
66.557315	1	3	5	7	9	16	88737189.
78.080804	3	4	5	6	14	20	285963050.
117.102215	3	5	6	15	16	20	856401896.
130.442207	3	5	6	14	16	20	**********
155.576834	3	4	5	14	15	20	508093686.
198.937523	3	5	6	14	15	16	**********

LIST OF REFERENCES

1. Driels, M.R., Swayze, W.E., and Potter, S.A., "Full Pose Calibration of a Robotic Manipulator Using a Coordinate Measuring Machine," Submitted for Publication, p. 1, 1991.

2. Swayze, W.E., "Modelling Experimental Procedures for Manipulator Calibration," Master's Thesis, Naval Postgraduate School, Monterey, California, December 1991, pp. 47-101.

3. Driels, M.R., Swayze, W.E., and Potter, S.A., "Full Pose Calibration of a Robotic Manipulator Using a Cooıdinate Measuring Machine," Submitted for Publication, p. 1, 1991.

4. Driels, M.R., "Using Passive End-Point Motion Constraints to Calibrate Kinematic Mechanisms," Submitted for Publication, 1991.

5. Swayze, W.E., "Modelling Experimental Procedures for Manipulator Calibration," Master's Thesis, Naval Postgraduate School, Monterey, California, December, 1991.

6. Paul, R.P., *Robot Manipulators: Mathematics, Programming, and Control*, The Massachusetts Institute of Technology Press, Chapters 1-3, 1982.

7. Swayze, W.E., "Modelling Experimental Procedures for Manipulator Calibration," Master's Thesis, Naval Postgraduate School, Monterey, California, Chapter II, December 1991.

8. Paul, R.P., *Robot Manipulators: Mathematics, Programming, and Control*, The Massachusetts Institute of Technology Press, Chapters 1-3, 1982.

9. Craig, J.J., *Introduction to Robotics, Mechanics and Control*, Addison-Wesley Publishing Company, 1989, Appendix B.

10. Denavit, J. and Hartenberg, R.S., "A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices", *ASME Journal of Applied Mechanics* (June 1955), p. 215-221.

11. Hayati, S. and Mirmirani, M., "A Software for Robot Geometry Estimation", SME Paper #MS84-1052, presented at Robots West Conference, Anaheim, California, November 1984.

12. Mooring, B.W., Roth, Z.S., and Driels, M.R., *Fundamentals of Manipulator Calibration*, p. 43, John Wiley and Sons, Inc., 1991.

13. Mooring, B.W., Roth, Z.S., and Driels, M.R., *Fundamentals of Manipulator Calibration*, pp 281,284, John Wiley and Sons, Inc., 1991.

14. Driels, M.R. and Pathre, U.S., "Significance of Observation Strategy on the Design of Robotic Calibration Experiments", *Journal of Robotic Systems* 7(2), 197-223 (1990).

15. Driels, M.R. and Pathre, U.S., "Significance of Observation Strategy on the Design of Robot Calibration Experiments" *Journal of Robotic Systems* 7(2), 197-223 (1990).

16. Driels, M.R. and Pathre, U.S., "Significance of Observation Strategy on the Design of Robot Calibration Experiments", *Journal of Robotic Systems* 7(2), 197-223 (1990).

17. Potter, S.A., "Full Pose and Partial Pose Calibration of a Six Degree of Freedom Robot Manipulator Arm," Master's Thesis, Naval Postgraduate School, Monterey, California, September, 1991.

18. Swayze, W.E., "Modelling Experimental Procedures of Manipulator Calibration," Master's Thesis, Naval Postgraduate School, Monterey, California, December, 1991.

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
Cameron Station
Alexandria, Virginia 22304-6145

2. Naval Engineering Curricular Office 1
Code 34
Department of Mechanical Engineering
Naval Postgraduate School
Monterey, California 93943-5000

3. Library, Code 0142 2
Naval Postgraduate School
Monterey, California 93943-5000

4. Department Chairman, Code ME 1
Department of Mechanical Engineering
Naval Postgraduate School
Monterey, California 93943-5000

5. Professor Morris Driels, Code ME/Dr 3
Department of Mechanical Engineering
Naval Postgraduate School
Monterey, California 93943-5000

6. CDR Ronald L. Edwards 3
P.O. Box 576
5 New Boston Road
Amherst, New Hampshire 03031